Solving problems in PKI

Overview

Conventional PKI requires the user to have the public privacy key of a recipient. This is not helpful when trying to communicate with a function rather than an individual. An alternative scheme for privacy could be adopted that solves this problem.

Introduction

Most people developing PKI systems use algorithms such as RSA or DSA to provide the digital signature by which the sender is authenticated. I say authenticated rather than identified, because if they have not been identified by a Certification Authority (CA) somewhere then you have to check their identity for yourself. Someone has to make the connection between a cryptographic key and the individual using it.

This works fine where you believe that the CA has checked on who the person actually is. Maybe the CA is also their employer, or a bank, or the government or some similar body that you think is likely to get it right.

However, the signature system works because the sender usually provides their 'digital certificate' along with the signed information so that it's easier to check it out. If they don't, then you have a problem figuring out who they might be because you have to know who to ask in advance.

The first problem

As you can now guess, digital signatures don't tell you about the unknown, only the known. Now if that is true for the digital signature, then what about privacy?

Well, privacy uses the same method as the digital signature, but working the other way round. You can't send a certificate along with the encrypted message because then anyone could read the message! So you have a problem. You need to have the recipients privacy certificate (it may be the same as their signing certificate, it depends on the scheme they're running, but that's getting even more complicated so we'll ignore it).

Now that's fine if you have already had something from them, but what about if you haven't? Now we're walking on thin ice. Somehow or other you need to be able to get hold of that certificate. That's because the way that RSA and DSA work you can't guess a value, you generate both keys at the same time and destroy the value that links them together.

So your first problem is that you need to be able to figure out who you are going to send something to in advance and make sure you have their key.

The second problem

This is a bit of a twist on the first problem. Very often you don't actually know who you are sending information to. What you do know is their functional responsibility. You don't know the individual in the tax office who is going to deal with your affairs, but you certainly don't want to share them with the whole Internet while you're doing it.

This is a sort of 'group' concept. You want to communicate with someone with the right functional responsibility inside a destination organization. Now when you get a reply from that functional responsibility you want to know who, as an individual, it is you are dealing with. You don't want some nameless function. But when you reply, you may need to deal with the individual, the function, or both.

Now this starts to get a bit confusing because you suddenly have an increasing number of people (and functions) that you are having to encrypt things for, and you have to keep track of it all, because you, as the sender, get to fix in stone who can read you information.

Solutions – good, bad and ugly

The ugly solution is classical PKI. You keep track of all the individuals. Hopefully there is a Directory that you can get hold of where you can pull down all the information you need on whoever you want to send things to. There is a downside to this approach. Whoever is running the Directory needs to be up to date and has to be very careful that what they publish is absolutely right, but at the same time does not publish to the outside world more than they want to declare about the organization.

A bad solution would be for the organization to publish a Directory with keys of functions for people outside to communicate with. This gets around a few problems in the ugly solution, like reducing the amount of information that gets published and reducing the amount that has to be protected since people change a lot faster than job functions. But it now means that lots of people inside the organization have to have the keys to be able to read information sent to that function or there has to be some kind of administration system that converts from one privacy key to another. That has to be heavily protected because it will store lots of keys that have to be kept out of the hands of hackers and internal staff.

A good solution could be developed based upon some published work by Clifford Cocks, about schemes for predicting public privacy keys.

An overview of the Clifford Cocks scheme

The name Clifford Cocks may not be immediately familiar to people outside the rather close community of cryptographers, but he is now credited as being the original inventor of asymmetric cryptography (schemes using two different keys), but since he was then (and still is) employed by the British national security agency GCHQ, his original work was not published because it was considered a national secret.

In the last couple of years he published a paper on a method for being able to predict the public privacy key of a recipient from published information referred to as ID-PKC. The mathematically inclined may like to look at the technical aspects of the scheme, which were published on the CESG web site www.cesg.gov.uk. (CESG or Communications- Electronic Security Group is the public facing arm of the British national security agency.)

The attractions of the scheme are considerable. It is possible to have a scheme which incorporates the date as one of its fields, as well as a function and the formal name of the organization being addressed. It is also possible to only issue the matching private key when it is required.

This is a stunning combination of capabilities for PKI. It gets around several problems that are otherwise quite difficult to solve.

If you incorporate the date then corporate privacy keys can be made self-updating without having any of the complications created by using Directory for rapidly changing keys. The scheme also means that a private privacy key only needs to be generated when it is required, it does not actually have to be stored all the time.

The public details of the scheme the corporate is using may be digitally signed so that external users can be certain the information they are using to generate the public privacy key is right.

It can also be made available to anyone inside the corporation who needs it rather than being personal to an individual. And since the value changes by the day no-one inside the corporation can compromise a significant amount of information if they do get access to a key when they should not. Even more importantly you can always get a copy of the key at a later date if it becomes necessary, so keys cannot be ‘lost’ if an individual leaves.

From the outsider’s point of view there are many good features for such an approach also. There is no need to find the name of an individual in the organization before you can send something in confidence. There is no concern about the ‘wrong’ people being able to read the information, or of having to maintain a catalogue of keys of all the people who might need to be able to read what is being sent. The sender can also send the information they used when generating the privacy public key because the scheme is not compromised by doing that. This helps if for any reason the data they used for predicting the privacy key was incorrect (perhaps they chose the wrong organizational function).

Conclusion

Public key cryptography is not restricted to a single approach of having to generate a pair of related keys simultaneously. Other schemes are mathematically possible, and such schemes offer important facilities that may be used to simplify some practical difficulties encountered when trying to implement schemes based solely on the RSA/DSA approach. There are no technical reasons why such schemes should not co-exist, and they may in fact be required when dealing with governmental bodies and others that have a need to work through functional titles rather than personal identities.